REMARKS

Claims 1-48 are pending in this application, with Claims 1, 11, 12, 22, 23, and 33-48 in independent form. Favorable reconsideration is requested.

STATEMENT OF THE SUBSTANCE OF THE INTERVIEW

Applicants acknowledge with appreciation the courtesies extended to their representatives, Douglas Sharrott and Peter Thurlow, by Examiner Rones during the telephone interview conducted on November 9, 2004.

In the Office Action, Claims 1-33 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,642,946 (Janes). During the interview, the features of Claims 1 and 9-11 were discussed in view of Janes. Applicants' representatives discussed how the claimed invention worked and how it differed from Janes. Applicants' representatives proposed amending those claims, as well as corresponding Claims 12, 20-23 and 31-33, to more clearly define the invention over Janes. No agreement was reached with respect to the claims.

Claims 1, 9-12, and 20-23 and 31-33, have been so amended, as shown above. Further, Claims 34-48 have been added to provide Applicants with a more complete scope of protection, reciting features discussed in the interview that are neither taught nor suggested by Janes. Support for these amendments can be found, for example, at paragraphs 19, 30 and 31, and Figs. 3A-3C and 5 of the application. It is submitted that independent Claims 1, 11, 12, 21, 22 and 33-45, together with the claims dependent thereon, are patentably distinct from Janes, as follows.

Janes relates to a livestock inventory and materials system with an interactive graphical user interface for viewing data. Each view contains a tree that a user may create by selecting one or more attributes of a pre-defined set of attributes (see Fig. 2A). For example, the "inventory view" 42 of Fig. 2A is a user's selection from the pre-defined list of attributes, and is used to construct a particular static tree view. (See also "inventory as of date" 43 and an "inventory to view" 44.)

In Janes, the user selects which attributes should be used to construct a tree. The hierarchical tree data structure is built by the Fill_Tree method in reference to parent-child data stored in a particular table, with each node having a parent node ID (see the Janes' specification at col. 7, line 52, to col. 9, line 18). Figures 7 and 8 of Janes illustrate how the tree is arranged as a static set of nodes, with each node having a number and a fixed parent ID. The tree is constructed with a level in the hierarchy for each selected attribute, and a node at a particular level for each pre-defined value that the level's attribute may have. The tree structure is subsequently saved in a table (see above). Subsequently, items are associated with the appropriate nodes in the tree.[1]

Accordingly, in Janes, the tree view is a function of the user's choice of attributes and nothing else. Unlike the present invention, in Janes the tree is defined statically and independently of the data underlying the items associated with the tree (as evidenced by the fact that the tree is stored in a separate table from the underlying data). In particular, in the present invention, the tree is a function of both the user's choice of attributes (defining the tree hierarchy) and of the underlying data (i.e., the values of the attributes). Thus, the tree of the present invention is not a static structure defined merely by the choice of attributes, as in Janes, but is advantageously a dynamic structure that is a function of the underlying data and that has the minimum number of nodes required to represent the data. This also means that the tree itself conveys useful information about the underlying data. Further unlike Janes, advantageously the nodes at a given hierarchy level are not pre-defined, but are driven by whatever values the items in the data-set have for the selected attribute, which is critical for a dynamic (data driven) tree structure.

1/ Janes, at col. 9, lines 11-18, lists a number of so-called "tree tools." Janes does not describe in any way how these work. In any event, they are all manual tools, requiring user action, and thus do not relate to methods and systems for automatically updating a tree.

Janes also discloses a drag and drop method for conducting a transaction. See col. 11, lines 25-28, of Janes. When the transaction is completed (indicated by dropping inventory on an intended target), the database is modified with the transaction data. "The source and target explorer display is refreshed to reflect the updated inventory status." See col. 11, lines 56-65, of Janes. Consequently, in Janes system, user action (performing a drag-and-drop transaction) is required to cause the display to refresh -- it is not automatic. Further, the "drag and drop" does not modify the tree – it merely changes the attribute of an item so that it becomes associated with another or new node on the tree. While this causes the re-association of data-items to nodes, it does not update or refresh the Janes' tree structure, because the latter is a static structure defined by the selection of attributes, and is thus unaffected by the attribute values of the underlying data.

There are addition differences between Janes and various aspects of the present invention. For example, in Janes, the attributes are fixed and are not calculated or derived from other attributes. Also in Janes, only one node can be selected at a time for displaying information on the right-side panel (see, for example, Figs. 2F-2J, 4A-4B, 6-9, and 10A-10B of Janes).

In one aspect of Applicants' invention, a user may select "any of the attributes" associated with the objects (all claims), and thus, the user is not limited in his selection to a predefined set of attributes as in Janes. In addition, Claims 34, 38 and 42 specifically recite that a user may select "any of the attributes in accordance with the user's preferences, wherein the set of available attributes available for selection is not pre-defined," which feature is neither taught nor suggested by Janes.

In another aspect of Applicants' invention as set forth in dependent Claims 10, 21, 32 and in independent Claims 36, 40 and 44, one or more of the attribute are calculated or derived from other attributes (see paragraph 19 of the specification, which describes examples of attributes that are calculated or derived from the values of other attributes). This feature is neither taught nor suggested by Janes.

In still another aspect of Applicants' invention as set forth in dependent Claims 9, 20 and 31 and independent Claims 35, 39 and 43, the user may simultaneously select "two or more nodes to operate upon or display all objects associated with the selected nodes," which feature is neither taught nor suggested by Janes.

In yet still another aspect of Applicants' invention as set forth in Claims 1, 12 and 23, the plurality of objects are automatically monitored to determine changes to the objects or their attributes. The tree is then updated based on those changes. Unlike Janes, no user action is required to cause the update of the tree. (See, for example, paragraphs 28, 30, and 31 of the specification.)

In yet still another aspect of the invention, as set forth in Claim 37, there is provided a method for creating a dynamic tree having a plurality of nodes and a plurality of objects associated therewith, each object having a plurality of attributes and each attribute having a value. This method includes the steps of selecting, by a user, any of the attributes in accordance with the user's preferences; creating the tree in accordance with the selected attributes and their respective values; and automatically updating the tree based on changes to one or more of the attribute values. Clearly, as discussed above, Janes does not teach or suggest a dynamic tree, or automatically updating a tree based on changes to the values of the attributes. Claims 41 and 45 are believed patentable over Janes for the same reasons.

New Claim 46 is directed to a method for creating a tree having a plurality of nodes and a plurality of objects associated therewith, each object having a plurality of intrinsic and derived attributes. This method includes the steps of selecting, by a user, any of the intrinsic and derived attributes in accordance with the user's preferences; creating the tree in accordance with the selected attributes and the values assigned to these attributes, within a universe of objects, wherein the tree requires the least number of nodes to represent all objects in the universe; and automatically monitoring the objects to determine changes to the objects or their attributes and updating the tree based on those changes to ensure that the tree requires the least number of nodes to represent all objects in the universe.

Janes does not teach or suggest that the attributes may be both inherent and derived, nor does it teach or suggest creating or updating a tree that requires the least number of nodes to represent all objects in the universe (which may be deemed to be an "optimal" tree.) According, Claim 46, and corresponding Claims 47-48, are believed patentable over Janes.

In still another aspect of the invention as set forth in Claims 11, 22 and 33, a plurality of objects are associated with a node, and each object has a plurality of attributes. The objects associated with any one of the nodes is a superset of objects associated with lower nodes. This is completely unlike Janes, in which the objects are located solely at the terminal (bottom) nodes. See col. 5, lines 61-63, of Janes. ("The bottom-most nodes (terminal nodes) of any branch are the only nodes, in that branch, that may contain data.") Further, unlike Janes, an attribute filter is applied to each lower node in successive fashion so that only those objects contained in a higher node that have an attribute matching the node attribute are displayed.

Thus, Janes' method is a bottom-up additive approach, where the objects displayed when a given node is selected are the *sum* of all objects associated with terminal nodes below the selected node in the tree. This aspect of Applicants' invention, however, in which objects may have multiple attributes and are associated with a node at every level of the tree (since the tree is built from the values of these attributes), uses a top-down approach, where the universe is successively attribute filtered as a user moves down the tree to restrict the set of associated objects. This difference is critical in ensuring that Applicants' invention maintains a dynamic tree – a new object added to the database may require more than one new node if it has unique values for more than one of the selected attributes. The top down filtering approach thus assures that there is no risk of an object being associated with a node that is not associated with the rest of the tree.

In view of the foregoing amendments and remarks, favorable reconsideration and early passage to issue of the present application is respectfully requested.

The undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should continue to be directed to our below listed address.

Respectfully submitted,

Douglas Sharrott
Attorney for Applicants
Registration No.: 39,832

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200